

## Лабораторная работа №1

### Методические указания

#### Списки в Lisp. Использование стандартных функций.

**Цель работы:** приобрести навыки использования списков и стандартных функций Lisp.

**Задачи работы:** изучить способ использования списков для фиксации информации, внутреннее представление одноуровневых и структурированных списков, методы их обработки с использованием базовых функций Lisp.

Отчет по лабораторной сдается в письменной форме по окончании работы.

#### Краткие теоретические сведения

Функциональное программирование ориентировано на символьную обработку данных. Предполагается, что любую информацию можно свести к символьной. Слово «символ» здесь близко к понятию «знак».

Базис Lisp образуют:

атомы, структуры, базовые функции, базовые функционалы.

#### Данные

Вся информация (данные и программы) в Lisp представляется в виде символьных выражений – **S-выражений**. По определению

**S-выражение ::= <атом> | <точечная пара>.**

**Элементарные значения структур данных:**

**Атомы:**

- **символы** (идентификаторы) – синтаксически – набор литер (букв и цифр), начинающихся с буквы;
- **специальные символы** – {T, Nil} (используются для обозначения логических констант);
- **самоопределимые атомы** – натуральные числа, дробные числа (например 2/3), вещественные числа, строки – последовательность символов, заключенных в двойные апострофы (например "abc");

Более сложные данные — **списки и точечные пары (структуры)** строятся из унифицированных структур – блоков памяти – бинарных узлов.

Запишем определения:

**Точечные пары ::= (<атом>, <атом>) | (<атом>, <точечная пара>) | (<точечная пара>, <атом>) | (<точечная пара>, <точечная пара>);**

**Список ::= <пустой список> | <непустой список>, где**

**<пустой список> ::= () | Nil,**

**<непустой список> ::= (<первый элемент>, <хвост>),**

**<первый элемент> ::= <S-выражение>,**

**<хвост> ::= <список>.**

**Синтаксически:**

любая структура (точечная пара или список) заключается в круглые скобки

(A, B) – точечная пара, (A) – список из одного элемента,

пустой список изображается как Nil или ();

непустой список по определению может быть изображен:



(A . (B . (C . (D ())))), допустимо изображение списка последовательностью атомов, разделенных пробелами — (A B C D).

Элементы списка могут, в свою очередь, быть списками (любой список заключается в круглые скобки),

например — (A (B C) (D (E))). Таким образом, синтаксически наличие скобок является признаком структуры — списка или точечной пары.

Любая непустая **структура Lisp** в памяти представляется списковой ячейкой, хранящей два указателя: на голову (первый элемент) и хвост — все остальное.

### Указания к выполнению работы

При выполнении лабораторной работы следует

- изобразить внутреннее представление списков, обратив внимание, что любому списку соответствует списковая ячейка. (рис. 1):

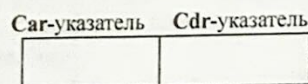


Рис.1. Представление целого числа

- изучить работу базовых функций в Lisp-е,
- изучить способы определения функций и их использование.

### Задание:

#### Лабораторная работа №1

##### Теоретические вопросы:

1. Элементы языка: определение, синтаксис, представление в памяти.
2. Особенности языка Lisp. Структура программы. Символ апостроф.
3. Базис языка Lisp. Ядро языка.

##### Практические задания:

1. Представить следующие списки в виде списочные ячеек:

' (open close halph)

' ((open1) (close2) (halph3))

'((one) for all (and (me (for you))))

' ((TOOL) (call))

' ((TOOL1) ((call2)) ((sell)))

' (((TOOL) (call)) ((sell)))

2. Используя только функции CAR и CDR, написать выражения, возвращающие

- 1) второй 2) третий 3) четвертый элементы заданного списка.

3. Что будет в результате вычисления выражений?

a) (CAADR ' ((blue cube) (red pyramid)))

c) (CADR ' ((abc) (def) (ghi)))

b) (CDAR ' ((abc) (def) (ghi)))

d) (CADDR ' ((abc) (def) (ghi)))

4. Напишите результат вычисления выражений и объясните как он получен:

(list 'Fred 'and 'Wilma)

(cons 'Fred '(and Wilma))

(list 'Fred '(and Wilma))

(cons 'Fred '(Wilma))

(cons Nil Nil)

(list Nil Nil)

(cons T Nil)

(list T Nil)

(cons Nil T)

(list Nil T)

(list Nil)

(cons T (list Nil))

(cons ' (T) Nil)

(list '(T) Nil)

(list ' (one two) ' (free temp))

(cons '(one two) '(free temp))

5. Написать лямбда-выражение и соответствующую функцию:

- Написать функцию (f ar1 ar2 ar3 ar4), возвращающую список:

((ar1 ar2) (ar3 ar4)).

- Написать функцию (f ar1 ar2), возвращающую ((ar1) (ar2)).

- Написать функцию (f ar1), возвращающую (((ar1))).

- Представить результаты в виде списочных ячеек.